Google

"entry address" protect   Search   Advanced Search
                                          Preferences

Web

Results **1 - 10** of about **944** for **"entry address" protect**. (0.43 seconds)

SIPP: Using Core Files
Users should note that in the 1996 Panel, the **entry address** ID is no longer ... That
is a change from earlier panels in which the **entry address** ID was key ...
www.sipp.census.gov/sipp/use_core_structure.html - 19k - Cached - Similar pages

SIPP Public Use Files
The **entry address** ID is the current address ID that a sample member ... in the
public use files to **protect** the confidentiality of survey respondents. ...
www.sipp.census.gov/sipp/pub_use.html - 17k - Cached - Similar pages

[PPT] BGP Countermeasures (Secure-BGP)
File Format: Microsoft Powerpoint 97 - View as HTML
Used to **protect** BGP from erroneous UPDATEs (authenticated but misbehaving
or ... DNS or Certificate server -- 1 **entry/address** block, 1 entry/AS, ...
www.net-tech.bbn.com/sbgp/IETF42.ppt - Similar pages

[PDF] Secure Execution Via Program Shepherding
File Format: PDF/Adobe Acrobat - View as HTML
sic block **entry address**. Transitions between blocks ... overhead of performing page
protection calls to prevent attacks against the system itself. ...
www.lcs.mit.edu/publications/ pubs/pdf/MIT-LCS-TM-625.pdf - Similar pages

[PDF] Microsoft PowerPoint - S-BGP-NDSS99.990204.ppt
File Format: PDF/Adobe Acrobat - View as HTML
Used to **protect** BGP from erroneous UPDATEs. (authenticated but misbehaving or
misconfigured ... DNS or Certificate server -- 1 **entry/address** block, 1 ...
www.isoc.org/isoc/conferences/ ndss/99/proceedings/slides/lynn.pdf - Similar pages

Win95.SK
... patches them and switches its DOS 16-bit code to **protect** mode 32-bit. ... the
virus writes its JMP_Virus patch directly to program's **Entry address**. ...
www.avp.ch/avpve/newexe/win95/sk.stm - 17k - Nov 7, 2005 -
Cached - Similar pages

Mechanism for accessing multiple virtual address spaces - Patent ...
... register 19 are added to obtain the required page table **entry address**. ... Storage
protect keys are used to provide intra-address-space **protection** for ...
www.freepatentsonline.com/4521846.html - 79k - Cached - Similar pages

Linux-Kernel Archive: Re: page fault scalability patch V10: [2/7 ...
swapin_readahead(**entry, address**, vma); ... spinlock held to **protect** against
concurrent faults in > - * multithreaded programs. ...
www.ussg.iu.edu/hypermail/linux/kernel/0410.1/2154.html - 17k -
Cached - Similar pages

taskLib
Also, a task can **protect** itself from deletion by taking a mutual-exclusion ... The
entry address entryPt is the address of the "main" routine of the task. ...
www.eelab.usyd.edu.au/tornado/ docs/vxworks/ref/taskLib.html - 39k -
Cached - Similar pages

Canon imageRUNNER 2230 - IKON Office Solutions
... plus a 200-**entry address** book for storing frequently dialed numbers. ... the
imageRUNNER 2230 offers these features to **protect** sensitive data: ...
www.ikon.com/products/Copiers/mono/Canon_iR2230.asp - 57k -
Cached - Similar pages

Goooooooooogle ▶

Result Page: **1** 2 3 4 5 6 7 8 9 10   **Next**

Google Desktop Search   9:30 AM

Free! Instantly find your email, files, media and web history. Download now.

"entry address" protect   Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

©2005 Google

Subscribe (Full Service)   Register (Limited Service, Free)   Login

Search: ◉ The ACM Digital Library   ○ The Guide

"entry address" protect

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **entry address** **protect**   Found **9,108** of **166,357**

Sort results by [relevance ▼]

Display results [expanded form ▼]

◈ Save results to a Binder

[?] Search Tips

☐ Open results in a new window

Try an **Advanced Search**
Try this search in **The ACM Guide**

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next
Best 200 shown
                                                    Relevance scale ☐ ▱ ▨ ▧ ▧

1 An implementation of a multiprocessing computer system
William B. Ackerman, William W. Plummer
January 1967 **Proceedings of the first ACM symposium on Operating System Principles**
Publisher: ACM Press
Full text available: 📄 pdf(626.32 KB)   Additional Information: full citation, abstract, references, citings, index terms

  A PDP-1 computer was donated (by the Digital Equipment Corporation) to the Electrical Engineering
  Department of the Massachusetts Institute of Technology in late 1961. In May, 1963 the first time-
  sharing system was operational. Since 1963 this PDP-1 has undergone substantial modifications (c.f.
  Appendix). Presently the machine has twelve thousand words (18-bit) of five microsecond memory
  arranged in pages of four thousand words. One of these pages is reserved for the system code ...

2 Session 14c: program library care and feeding: Program library maintenance and monitoring
John R. Ehrman
November 1976 **Proceedings of the 4th annual ACM SIGUCCS conference on User services**
Publisher: ACM Press
Full text available: 📄 pdf(551.38 KB)   Additional Information: full citation, abstract

  This memorandum describes the organization of the SLAC Triplex Program Library, and the methods to
  be used for its maintenance. Also described is an important maintenance tool, the Library Monitoring
  System.

3 Mondrian memory protection
Emmett Witchel, Josh Cates, Krste Asanović
October 2002 **ACM SIGPLAN Notices , ACM SIGARCH Computer Architecture News , ACM SIGOPS
          Operating Systems Review , Proceedings of the 10th international conference on
          Architectural support for programming languages and operating systems ASPLOS-X,**
          Volume 37 , 30 , 36 Issue 10 , 5 , 5
Publisher: ACM Press
Full text available: 📄 pdf(1.53 MB)   Additional Information: full citation, abstract, references, citings

  Mondrian memory protection (MMP) is a fine-grained protection scheme that allows multiple protection
  domains to flexibly share memory and export protected services. In contrast to earlier page-based
  systems, MMP allows arbitrary permissions control at the granularity of individual words. We use a
  compressed permissions table to reduce space overheads and employ two levels of permissions caching
  to reduce run-time overheads. The protection tables in our implementation add less than 9% overhead
  to ...

4 Dynamic Supervisors - their design and construction
D. H. R. Huxtable, M. T. Warwick
January 1967 **Proceedings of the first ACM symposium on Operating System Principles**
Publisher: ACM Press
Full text available: 📄 pdf(719.81 KB)   Additional Information: full citation, abstract, citings, index terms

  The paper demonstrates the technology necessary to bring the facilities of Supervisor construction and
  modification to the level at which a user can, without a great deal of research and analysis modify his
  installation's Operating System The Supervisor is seen to be a set of processes linked by a formalised
  control mechanism.

5 Lightweight remote procedure call
Brian N. Bershad, Thomas E. Anderson, Edward D. Lazowska, Henry M. Levy
February 1990 **ACM Transactions on Computer Systems (TOCS)**, Volume 8 Issue 1

Publisher: ACM Press
Full text available: pdf(1.60 MB)   Additional Information: full citation, abstract, references, citings, index terms, review

Lightweight Remote Procedure Call (LRPC) is a communication facility designed and optimized for communication between protection domains on the same machine. In contemporary small-kernel operating systems, existing RPC systems incur an unnecessarily high cost when used for the type of communication that predominates—between protection domains on the same machine. This cost leads system designers to coalesce weakly related subsystems into the same protection domain, trading safety for ...

6 Reliability experience with Chi/OS
W. C. Lynch, J. W. Langner, M. S. Schwartz
April 1975   **ACM SIGPLAN Notices , Proceedings of the international conference on Reliable software**, Volume 10 Issue 6
Publisher: ACM Press
Full text available: pdf(560.84 KB)   Additional Information: full citation, abstract, references, index terms

The Chi/OS operating system, the latest large scale software effort of Chi Corporation, has had an excellent reliability record since its installation in November, 1973. Although the system design is vital to the reliability of Chi/OS, several environmental factors are equally vital. After a brief presentation of the substantial work load supported by Chi/OS, this paper deals with those environmental factors which contribute to the reliability of the software.

Keywords: Design environment, Development environment, INCLUDE mechanism, Implementation language, Operational environment, Small group of highly skilled systems programmers, Systems programming apprenticeship

7 Workshop on architectural support for security and anti-virus (WASSA): A reliable return address stack: microarchitectural features to defeat stack smashing
Dong Ye, David Kaeli
March 2005   **ACM SIGARCH Computer Architecture News**, Volume 33 Issue 1
Publisher: ACM Press
Full text available: pdf(339.40 KB)   Additional Information: full citation, abstract, references, index terms

Buffer overflow vulnerability is one of the most common security bugs existing in today's software systems. In this paper, we propose a microarchitectural design of a return address stack aiming to detect and stop stack smashing. This approach has been used in other proposals to guard against buffer overflow vulnerabilities. Our contribution is a design that handle multipath execution, speculative execution, abnormal control flow, and extended call depth. Our solution makes no assumption about t ...

8 A search algorithm and data structure for an efficient information system
Shou-chuan Yang
September 1969 **Proceedings of the 1969 conference on Computational linguistics**
Publisher: Association for Computational Linguistics
Full text available: pdf(1.54 MB)   Additional Information: full citation, abstract, references

This paper describes a system for information storage, retrieval, and updating, with special attention to the search algorithm and data structure demanded for maximum program efficieny. The program efficiency is especially warranted when a natural language or a symbolic language is involved in the searching process.The system is a basic framework for an efficient information system. It can be implemented for text processing and document retrieval; numerical data retrieval; and for handling of la ...

9 The performance of μ-kernel-based systems
Hermann Härtig, Michael Hohmuth, Jochen Liedtke, Sebastian Schönberg
October 1997 **ACM SIGOPS Operating Systems Review , Proceedings of the sixteenth ACM symposium on Operating systems principles SOSP '97**, Volume 31 Issue 5
Publisher: ACM Press
Full text available: pdf(2.02 MB)   Additional Information: full citation, references, citings, index terms

10 A study of initialization in Linux and OpenBSD
Catherine Dodge, Cynthia Irvine, Thuy Nguyen
April 2005   **ACM SIGOPS Operating Systems Review**, Volume 39 Issue 2
Publisher: ACM Press
Full text available: pdf(2.02 MB)   Additional Information: full citation, abstract, references, index terms

The code that initializes a system can be notoriously difficult to understand. In secure systems, initialization is critical for establishing a starting state that is secure. This paper explores two architectures used for bringing an operating system to its initial state, once the operating system gains

control from the boot loader. Specifically, the ways in which the OpenBSD and Linux operating systems handle initialization are dissected.

11 The architecture of the SPERRY UNIVAC 1100 series systems

B. R. Borgerson, M. D. Godfrey, P. E. Hagerty, T. R. Rykken

April 1979 **Proceedings of the 6th annual symposium on Computer architecture**

Publisher: ACM Press

Full text available: pdf(841.19 KB)   Additional Information: full citation, abstract, references, citings, index terms

This paper presents an overview of the architecture of the SPERRY UNIVAC® 1100 Series systems. The principal topics are instruction and data formats, main storage and addressing, process management, and I/O.

12 Eliminating the address translation bottleneck for physical address cache

Tzi-cker Chiueh, Randy H. Katz

September 1992 **ACM SIGPLAN Notices , Proceedings of the fifth international conference on Architectural support for programming languages and operating systems ASPLOS-V,** Volume 27 Issue 9

Publisher: ACM Press

Full text available: pdf(1.28 MB)   Additional Information: full citation, references, citings, index terms

13 An appraisal of the Atlas supervisor

Derrick Morris, Frank H. Sumner, Michael T. Wyld

January 1967 **Proceedings of the 1967 22nd national conference**

Publisher: ACM Press

Full text available: pdf(818.62 KB)   Additional Information: full citation, abstract, references, citings, index terms

This report presents the performance of the Supervisor System used on the Atlas Computer at Manchester University, and describes some of the changes made as a result of our experience with the system. Although the machine is used jointly by I.C.T. Computing Service Division and the University Computing Service (U.C.S.), the figures presented are derived mainly from the U.C.S. use of the machine. We begin with an outline of the system and then describe its main sections in some detail. The i ...

14 The Honeywell Modular Microprogram Machine: M3

E. Douglas Jensen, Richard Y. Kain

March 1977 **ACM SIGARCH Computer Architecture News , Proceedings of the 4th annual symposium on Computer architecture ISCA '77**, Volume 5 Issue 7

Publisher: ACM Press

Full text available: pdf(883.59 KB)   Additional Information: full citation, abstract, references, citings, index terms

M3 is intended for research into unconventional special purpose stored program elements of computer systems (for example, a distributed computer Bus Interface Unit). The principal requirements for such a machine are flexibility and modularity. M3 consists of an application independent Kernel Machine to which application-dependent Functional Modules are attached. The Kernel Machine is vertically microprogrammed; it includes highly capable microinstru ...

Keywords: Computer architecture, Microprogramming, Structured programming

15 An architectural framework for migration from CISC to higher performance platforms

Gabriel M. Silberman, Kemal Ebcioğlu

August 1992 **Proceedings of the 6th international conference on Supercomputing**

Publisher: ACM Press

Full text available: pdf(2.04 MB)   Additional Information: full citation, abstract, references, citings, index terms

We describe a novel architectural framework that allows software applications written for a given Complex Instruction Set Computer (CISC) to migrate to a different, higher performance architecture, without a significant investment on the part of the application user or developer. The framework provides a hardware mechanism for seamless switching between two instruction sets, resulting in a machine that enhances application performance while keeping the same program behavior (from a user per ...

16 A fast string-matching algorithm for network processor-based intrusion detection system

Rong-Tai Liu, Nen-Fu Huang, Chih-Hao Chen, Chia-Nan Kao

August 2004 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 3 Issue 3

Publisher: ACM Press

Full text available: pdf(571.00 KB)   Additional Information: full citation, abstract, references, index terms

Network intrusion detection systems (NIDSs) are one of the latest developments in security. The

matching of packet strings against collected signatures dominates signature-based NIDS performance. Network processors are also one of the fastest growing segments of the semiconductor market, because they are designed to provide scalable and flexible solutions that can accommodate change quickly and economically. This work presents a fast string-matching algorithm (called FNP) over the network proces ...

Keywords: Intrusion detection, network, pattern matching, processor

---

17 [Applications of formal methods: Using build-integrated static checking to preserve correctness invariants](#)
Hao Chen, Jonathan S. Shapiro
October 2004   **Proceedings of the 11th ACM conference on Computer and communications security**
Publisher: ACM Press
Full text available: pdf(208.82 KB)   Additional Information: full citation, abstract, references, index terms

A key missing link in the creation of secure and robust systems is finding a ost effective way to demonstrate and preserve correspondence between a software design and its implementation. This paper explores the use of software model checking techniques to validate selected design invariants in the EROS operating system kernel. Several global consistency policies in the EROS kernel can be expressed as finite state automata. Using the MOPS static hecker, we have been able to validate the EROS ...

Keywords: EROS, MOPS, assurance, model hecking, security, static analysis, verification

---

18 [A snapshot differential refresh algorithm](#)
Bruce Lindsay, Laura Haas, C. Mohan, Hamid Pirahesh, Paul Wilms
June 1986   **ACM SIGMOD Record , Proceedings of the 1986 ACM SIGMOD international conference on Management of data SIGMOD '86**, Volume 15 Issue 2
Publisher: ACM Press
Full text available: pdf(827.36 KB)   Additional Information: full citation, abstract, references, citings, index terms

This article presents an algorithm to refresh the contents of database snapshots. A database snapshot is a read-only table whose contents are extracted from other tables in the database. The snapshot contents can be periodically refreshed to reflect the current state of the database. Snapshots are useful in many applications as a cost effective substitute for replicated data in a distributed database system. When the snapshot contents are a simpl ...

---

19 [VAX DEBUG: an interactive, symbolic, multilingual debugger](#)
Bert Beander
March 1983   **ACM SIGSOFT Software Engineering Notes , ACM SIGPLAN Notices , Proceedings of the symposium on High-level debugging SIGSOFT '83**, Volume 8 , 18 Issue 4 , 8
Publisher: ACM Press
Full text available: pdf(655.76 KB)   Additional Information: full citation, abstract, references

Digital Equipment Corporation's VAX-11 Debugger, usually called VAX DEBUG or simply DEBUG, is an interactive, symbolic, and multilingual debugger which runs on the VAX-11 series of computers under the VMS operating system. The following gives an overview of VAX DEBUG and examines how it solves some of the problems inherent in the design of any such debugger. Particular attention is paid to how its command language is designed, how it distinguishes between addresses and values in command input, h ...

---

20 [Binary translation and architecture convergence issues for IBM system/390](#)
Michael Gschwind, Kemal Ebcioğlu, Erik Altman, Sumedh Sathaye
May 2000   **Proceedings of the 14th international conference on Supercomputing**
Publisher: ACM Press
Full text available: pdf(1.44 MB)   Additional Information: full citation, abstract, references, index terms

We describe the design issues in an implementation of the ESA/390 architecture based on binary translation to a very long instruction word (VLIW) processor. During binary translation, complex ESA/390 instructions are decomposed into instruction "primitives" which are then scheduled onto a wide-issue machine. The aim is to achieve high instruction level parallelism due to the increased scheduling and optimization opportunities which can be exploited by binary translation software ...

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next